Front. Eng. Manag. 2025, 12(4): 1124–1140
https://doi.org/10.1007/s42524-025-4133-z

RESEARCH ARTICLE

Pingye TIAN, Qing YANG, Yingxin BI

Clustering new product development projects from the perspective of knowledge management

Abstract Knowledge transfer among New Product Development (NPD) projects is beneficial for reducing project duration and promoting technological innovation. To support effective knowledge transfer, we propose a clustering method for NPD projects based on similarity, integrating both structural and attribute similarities. First, to measure project structural similarity, we analyze both direct and indirect knowledge transfer relationships among project activities using the dependency structure matrix (DSM). Second, we measure project attribute similarity by calculating knowledge increments derived from sequential and iterative development processes. Finally, we apply a hierarchical clustering method to group similar projects, forming different programs. An industrial example is provided to demonstrate the proposed model. The results show that clustering projects into programs can enhance multi-project management by reducing coordination time for knowledge transfer within each program. Additionally, this approach provides some new insights, including quantifying project similarity based on knowledge transfer and understanding the influence of structural and attribute similarities on multi-project management.

Keywords multi-project management, new product development (NPD), knowledge management, clustering, dependency structure matrix (DSM)

Received Sep. 1, 2024; revised Nov. 26, 2024; accepted Dec. 8, 2024

Pingye TIAN, Qing YANG (✉)
School of Economics and Management, University of Science and Technology Beijing, Beijing 100083, China
E-mail: yangqing@manage.ustb.edu.cn

Yingxin BI
Shougang Institute of Talent Development (Party School), Beijing 100144, China

This research was supported by the National Natural Science Foundation of China (Grant Nos. W2441021 and 72271022).

1 Introduction

Efficiently leveraging knowledge across new product development (NPD) projects enables enterprises to gain competitive advantages. Currently, many enterprises face the challenge of concurrently and efficiently managing multiple projects (Liu et al., 2024). Knowledge transfer among NPD projects has the potential to shorten development cycles and drive technological innovation (Korhonen et al., 2016; Zhang et al., 2016). For example, the technology and experience accumulated by Apple during the iPhone project were successfully transferred to the iPad project. Features of the iOS operating system and user interface design were reused in the development of the iPad. This knowledge transfer helped Apple gain a competitive advantage in the market. Consequently, knowledge transfer among projects plays a crucial role in ensuring program success (Zhou et al., 2022). Organizations require project management methodologies grounded in a knowledge-based perspective to efficiently coordinate multiple projects, teams and activities, thereby ensuring the achievement of expected outcomes.

Identifying similar projects and clustering them into programs can facilitate knowledge transfer among projects and improve enterprise performance (Abbasi et al., 2020). Similar projects typically share similar knowledge related to development activities and products. Therefore, examining project characteristics is a key approach to identifying project similarity. Klessova et al. (2020) proposed that knowledge-based activity dependencies (e.g., knowledge transfer) are critical for the relationship among projects in a program. However, previous approaches for measuring project similarity were mainly based on project pay item compositions (Qiao et al., 2019) or the part/product networked operations sequence (Navaei and Elmaraghy, 2018). Existing studies do not consider knowledge transfer when measuring project similarity.

Multi-projects can be viewed as a network with two important features: a network structure that represents relationships among projects and a node attribute that describes the features of each project (Ou et al., 2022; Yang et al., 2022). Therefore, we analyze the project similarity from two dimensions: structural and attribute similarities. First, existing approaches to measuring project structural similarity are mainly based on the relationship of activity sequences (Navaei and Elmaraghy, 2016), but they ignored knowledge transfer in NPD projects, especially the impact of knowledge transfer related to indirect processes and activity iterations on project structural similarity (Qiao et al., 2019). Activity iteration refers to the repetition (or rework) of activities represented by feedback loops or cycles. Iteration is a fundamental feature of NPD projects, as it helps produce satisfactory deliverables while increasing the risk of rework in upstream activities (Yang et al., 2014). This study proposes a method for measuring project structural similarity based on knowledge transfer relationships. This includes direct relationship (i.e., common activities directly linking projects) and indirect relationship (i.e., intermediate activities in one project link the common activities). We assume that common knowledge between projects can facilitate knowledge transfer and increase program management efficiency.

Second, as the project progresses, each project's knowledge is continually accumulated, representing sequential-based knowledge increments. Meanwhile, upstream activities receive feedback from downstream activities, representing the iteration-based knowledge increments. The reuse and sharing of common knowledge between similar design activities across different projects significantly influence project node attribute similarity. Therefore, project node attribute similarity can be measured by the sequential-based and iteration-based knowledge increments of each project's activities. This comprehensive approach (i.e., integrating project structure and attribute similarities) considers the process of applying new external knowledge and sharing common knowledge across projects. Knowledge transfer and sharing can save development time and reduce project costs (Lin and Wang, 2019).

Therefore, we present a knowledge-based synthesis approach for measuring project similarity, which incorporates both structural and attribute similarities within a multi-project network (Fig. 1).

Design structure matrix (DSM) is a tool for analyzing the coupling dependencies between elements in NPD projects (Browning, 2016). Therefore, we use DSM (Design Structure Matrix, recently known as Dependency Structure Matrix or Dependency and Structure Modeling) to visualize the intra-project knowledge transfer relationship and knowledge increments. Existing studies on process DSM have focused on representing the activity sequence in an individual project (Browning, 2016). However, they ignored multi-project knowledge transfer. In this study, we measure multi-project similarity through two types of coefficients: (1) structural similarity coefficients from intra-project knowledge transfer relationships due to the common activity sequence, and (2) attribute similarity coefficients from the perspective of knowledge increments.

Clustering projects helps manage a multi-project architecture, where each project shares common knowledge to handle similar activities, thereby increasing knowledge transfer and improving overall project performance. A substantial amount of literature has been developed on clustering in project management. In the existing literature, K-means (Abdolvand et al., 2015), hierarchical clustering algorithm (Daie and Li, 2016), two-stage clustering criteria (Yang et al., 2014; Bi et al., 2020), and spectral clustering method (Yang et al., 2022) are the most popular approaches. Bi et al. (2020) proposed a model to measure the connection strength among projects based on knowledge transfer. They suggested a two-stage clustering method that grouped projects with strong knowledge connection strength to form a program instead of forming similar projects into a program. However, it ignored the integrated influence of project networks and failed to represent the attributes of each project within the network.

The hierarchical clustering method offers several advantages, including the absence of the need to specify the number of clusters in advance, better visualization and interpretability, and an easily identifiable cut-tree point for determining the number of clusters (Patriarca et al., 2023; Vali et al., 2022). Consequently, this study applies a hierarchical clustering algorithm for clustering projects based on integrated similarity coefficients (i.e.,

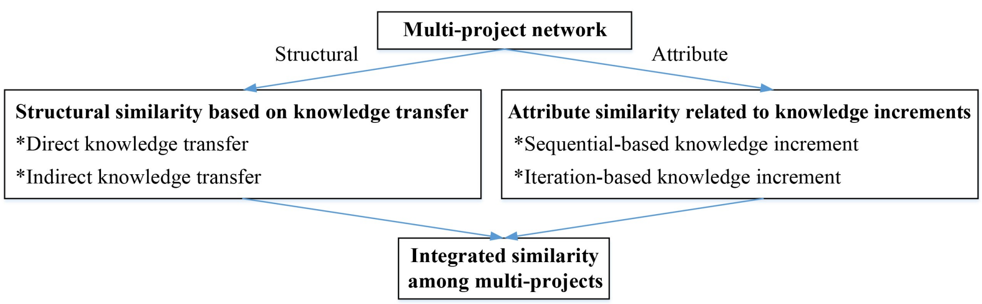


Fig. 1 Measuring integrated similarity among NPD multiple projects.

both structural similarity and attribute similarity). A detailed description of symbols used in this paper is shown in Appendix A.

This study makes three key contributions: (1) It contributes to the NPD project knowledge management literature by analyzing the impact of sequence and iterative processes among activities on knowledge transfer and increment. (2) It helps project managers in evaluating the knowledge-based similarity among projects and support them in clustering multiple projects to facilitate knowledge transfer. (3) It proposes a more sophisticated clustering method for NPD projects than previous studies by accounting for structural and attribute similarities among projects.

2 Literature review

2.1 Multi-project management

The goal of traditional single-project management is to maximise the performance of a single deliverable within a given range of costs, quality and time. It has been criticized for overlooking the reuse of critical elements such as technology, components and market knowledge (Haug, 2023). Consequently, organizations are faced with the need to manage multiple projects effectively based on their unique characteristics (Chen et al., 2022). Multiple projects should be managed in parallel according to technology and market-related interdependency relationships in a low-resource environment (Pan et al., 2022; Bai et al., 2022). For example, Korhonen et al. (2016) proposed that increasing component commonality and reducing the number of different components used between projects can reduce project costs. Therefore, multi-project management theory emphasizes the importance of program and portfolio management from the perspectives of technology transfer and knowledge management.

2.2 Knowledge transfer and increment in NPD projects

Several researchers have analyzed NPD processes based on knowledge transfer. For example, Stock et al. (2021) emphasized the importance of knowledge transfer in NPD and found that it plays a mediating role between uncertainty and performance. Knowledge transfer, occurring among different projects within a program through similar activities, enables the reuse of relevant knowledge across the entire program (Frank and Ribeiro, 2014). Due to the complex and diversified characteristics of NPD multi-projects, a wealth of project knowledge will accumulate and flow continuously in the project contact network composed of multiple participants. Argote and Fahrenkopf (2016) proposed that an enterprise with rich knowledge resources can apply them to multiple projects. Haug (2023) identified the factors that influence knowledge transfer in NPD projects. On the other hand, some literature also studied knowledge increments in NPD projects. Zhang and Min (2022) argued that greater knowledge increments and accumulation facilitate NPD success and enhance innovation performance. Santoro et al. (2006) analyzed knowledge increments and knowledge transfer across organizations in a project's progress.

In summary, although existing literature focuses on knowledge management in NPD projects, it has not analyzed the influence of knowledge transfer and knowledge increments on project similarity. Consequently, there exists an opportunity for enhancement by assessing project similarity through the lens of knowledge transfer and knowledge increments, enabling more efficient clustering of similar projects.

2.3 Similarity and clustering of NPD projects

Knowledge is considered a strategic resource for enterprises because it provides a basis for long-term competitive advantage (Xu et al., 2022). Therefore, program management success is inseparable from proper knowledge management. However, existing methods of measuring NPD project similarity ignore knowledge-based behaviors such as knowledge transfer. Qiao et al. (2019) proposed a method for measuring the similarity of different projects based on their pay-item compositions. Traditional methods of measuring activity similarity are primarily based on the number of joint activities, known as Jaccard similarity (Xu et al., 2022). However, this approach cannot judge the same number of common activities but different activity sequences. Dissimilar project activity sequences may lead to redundant personnel and machine idleness considering the complexity of multi-projects. Navaei and Elmaraghy (2016) proposed a similarity coefficient based on a networked activity sequence. However, they did not consider the impact of indirect and iteration-based knowledge transfer relationships on the similarity coefficient.

Various clustering algorithms for project management have been proposed in the literature. Yang et al. (2014) proposed a two-stage clustering criterion to measure team similarity in a product development project. Furthermore, Yang et al. (2022) used the spectral clustering method to cluster high-similarity teams. Abdolvand et al. (2015) analyzed project performance from customers' perspectives using the K-means clustering algorithm. However, these studies have mainly focused on comparing multiple clustering methods, and few have considered the visualization effect of clustering results. Daie and Li (2016) managed multi-product variety through a hierarchical clustering algorithm with good visualization and interpretable effect. This study uses a hierarchical clustering method to group projects with a high degree of similarity into a program.

In summary, clustering projects based on the similarity of activity sequences offers several advantages, such as minimizing machine idle time and reducing overall costs. However, it is also crucial to consider the iterative and indirect knowledge transfer among shared project activities.

3 Similarity-based clustering method for NPD projects from the knowledge management perspective

3.1 The framework of the proposed method

To improve knowledge management efficiency in NPD, we propose a clustering method based on project similarity. This method considers both structural similarity, due to knowledge transfer, and attribute similarity, stemming from knowledge increments. The similarity coefficients between projects are used as inputs for an algorithm that clusters multiple NPD projects. Our approach involves three steps (Fig. 2).

(1) Measuring structural similarity among projects. First, we calculate the structural similarity from direct knowledge transfer by analyzing sequential and iterative relationships between activities. Next, we assess the structural similarity from indirect knowledge transfer by examining the number of common intermediary activities.

(2) Measuring attribute similarity among projects. First, we calculate the knowledge increments generated through the sequential execution of activities. Next, we assess the knowledge increments arising from iterative processes, considering iteration probability and impact strength. Finally, we use Euclidean distance to evaluate attribute similarity between projects from the perspective of knowledge increments.

(3) Clustering analysis for NPD projects. First, we combine structural similarity and attribute similarity to build an overall similarity coefficient among projects. Then, the overall similarity coefficients are input into a hierarchical clustering algorithm to segment multiple projects. Finally, we evaluate the clustering results using the Cophenetic Correlation Coefficient and the Silhouette Index.

3.2 Measuring project structural similarity based on knowledge transfer

The sequence of activities establishes the intra-project knowledge transfer relationship, as knowledge from upstream activities can serve as input for downstream activities (Bashir et al., 2022). In this process, upstream activities transfer knowledge, then downstream activities receive it. However, existing similarity coefficients for networked activity sequences do not account for iteration (Navaei and Elmaraghy, 2016). Furthermore, traditional methods consider the impact of knowledge transfer relationships only when common activities between projects are adjacent, overlooking those that are not adjacent (i.e., indirect connections).

Therefore, we extend the networked activity sequence method and propose a structural similarity coefficient to model intra-project knowledge transfer relationships. This approach incorporates both the iterative characteristics between activities and the impact of indirect connections.

3.2.1 Intra-project direct knowledge transfer

Intra-project knowledge transfer refers to the exchange of information and experiences between upstream and downstream activities within a single project (Cao et al., 2024). We use the dependency structure matrix (DSM) to describe the intra-project knowledge transfer relationship among project activities. The process architecture of the DSMs of two projects is a square matrix composed of four parts, where the diagonal cells represent the activities in a project (Fig. 3). For example, A_1, A_2, A_4, and A_6 are common activities in project 1 (P_1) and project 2 (P_2). The off-diagonal entries in the DSM indicate direct relationships between activities. The sub-diagonal entries represent sequential knowledge transfers, while super-diagonal entries represent iteration-based knowledge transfers.

NPD can be seen as a creative and innovative process driven by iteration, which produces a satisfactory deliver-

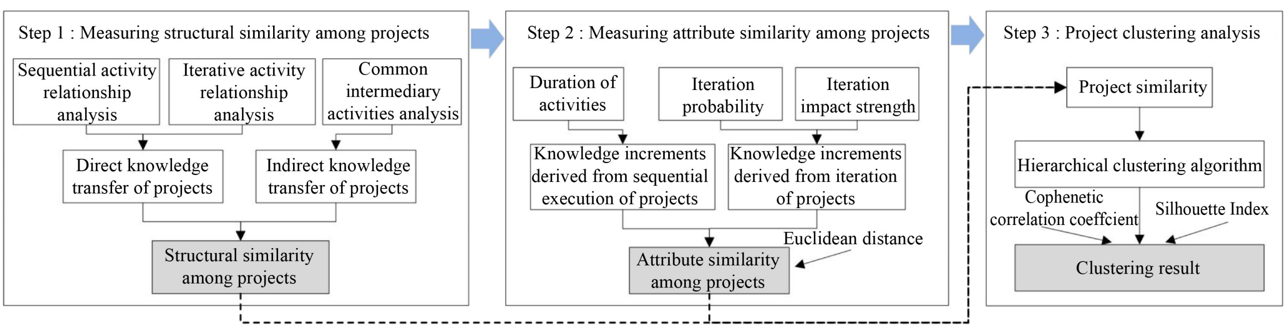


Fig. 2 The framework of the proposed method.

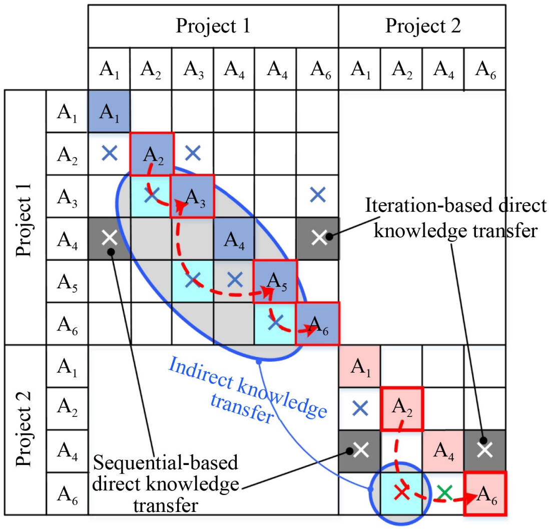


Fig. 3 An example showing knowledge transfer of intra-project via DSM.

able while causing upstream activities to undergo rework. Upstream activities are repeated in response to feedback from downstream activities. Activity iteration can enhance the quality of a project's ultimate deliverable. Multiple iterations can also accelerate the project schedule when appropriate iterative learning is applied. Thus, activity iteration is a key factor influencing structural similarity. This paper assumes that if the sequence and iterative processes of activities within two projects are similar, then the two projects are also similar from the perspective of knowledge transfer.

The connection between projects P_I and P_J can be calculated using the DSM model. We use $P_{I}\_DSM(i,j)$ to represent whether there is a direct knowledge transfer between activities i and j in project P_I and $P_{I}\_DSM(i,j)=1$ when direct knowledge transfer exists. Therefore, if there is direct knowledge transfer between activities i and j for both coupled projects P_I and P_J, then $P_{I}\_DSM(i,j)\cdot P_{J}\_DSM(i,j)=1$. As knowledge transfer is a directional process, $\sum_{j_c=1}^{m}\max(\sum_{i_c=1}^{m} P_{I}\_DSM(i_c,j_c), \sum_{i_c=1}^{m} P_{J}\_DSM(i_c,j_c))$ can represent accepting knowledge relationships, aggregating the maximum sum of rows between projects P_I and P_J of common activities (e.g., A_1, A_2, A_4 and A_6 of projects P_1 and P_2), respectively. Similarly, $\sum_{i_c=1}^{m}\max(\sum_{j_c=1}^{m} P_{I}\_DSM(i_c,j_c), \sum_{j_c=1}^{m} P_{J}\_DSM(i_c,j_c))$ is the sending of knowledge relationships, aggregating the maximum sum of columns between projects P_I and P_J of common activities, respectively.

Further, the structural similarity coefficient among projects based on the intra-project direct knowledge transfer relationship can be measured using Eq. (1):

$$S_{kt\_D}(P_I,P_J)=\frac{\sum_{i=1}^{n}\sum_{j=1}^{n} P_{I}\_DSM(i,j)\cdot P_{J}\_DSM(i,j)+\sum_{j=1}^{n}\sum_{i=1}^{n} P_{I}\_DSM(i,j)\cdot P_{J}\_DSM(i,j)}{\sum_{j_c=1}^{m}\max(\sum_{i_c=1}^{m} P_{I}\_DSM(i_c,j_c),\sum_{i_c=1}^{m} P_{J}\_DSM(i_c,j_c))+\sum_{i_c=1}^{m}\max(\sum_{j_c=1}^{m} P_{I}\_DSM(i_c,j_c),\sum_{j_c=1}^{m} P_{J}\_DSM(i_c,j_c))}, \tag{1}$$

where n represents the number of total activities of the multi-project, m is the number of common activities between projects, i_c and j_c are the rows and columns of common activities between projects P_I and P_J, respectively.

3.2.2 Intra-project indirect knowledge transfer

According to Eq. (1), the intra-project knowledge transfer relationship is dissimilar when common activities between projects are not adjacent. However, the similarity coefficient does not account for the two indirect knowledge transfer relationships between projects P_1 and P_2 (Fig. 3). The shadow represents intermediate activities (A_3 and A_5) between A_2 and A_6 in P_1. For example, activities A_2 and A_6 are common in projects P_1 and P_2, directly linked in project P_2, while part of the process in P_1 is $A_2 \rightarrow A_3 \rightarrow A_5 \rightarrow A_6$; that is, activities A_2 and A_6 are linked through intermediate activities A_3 and A_5. Hence, to supplement the aforementioned similarity coefficient, we present the intra-project indirect knowledge transfer relationship related to the number of common intermediary activities.

Therefore, the indirect connection between projects P_I and P_J can be calculated. The relationship between the intermediate activities of common activities and indirect knowledge transfer relationships is inversely proportional. The greater the number of activities between non-adjacent common activities, the weaker the impact on knowledge relationships. As the number of knowledge transfer edges connecting non-adjacent common activities between projects is the value of intermediate activities plus one, $N_m(i_m, j_m)$ is used to represent the number of intermediate activities when projects P_I and P_J have an indirect knowledge transfer relationship between common activities. $\frac{1}{\sum_{i_m=1}^{m}\sum_{j_m=1}^{m}(N_m(i_m,j_m)+1)}$ indicates the indirect impact of sending knowledge relationships and $\frac{1}{\sum_{j_m=1}^{m}\sum_{i_m=1}^{m}(N_m(i_m,j_m)+1)}$ indicates the impact of indirectly accepting knowledge relationships. The former is based on the number of intermediate activities calculated by the column in DSM. The latter is based on the number of intermediate activities calculated by the row in DSM.

Based on Eq. (1), the similarity coefficient influenced by the indirect knowledge transfer relationship can be calculated using Eq. (2), the proportion of the indirect knowledge transfer relationship relative to the aggregated maximum sum of rows and columns between projects P_I and P_J of common activities:

$$S_{kt\_ID}(P_I,P_J)=\frac{\frac{1}{\sum_{i_m=1}^{m}\sum_{j_m=1}^{m}(N_m(i_m,j_m)+1)}+\frac{1}{\sum_{j_m=1}^{m}\sum_{i_m=1}^{m}(N_m(i_m,j_m)+1)}}{\sum_{j_c=1}^{m}\max(\sum_{i_c=1}^{m}P_I\_DSM(i_c,j_c),\sum_{i_c=1}^{m}P_J\_DSM(i_c,j_c))+\sum_{i_c=1}^{m}\max(\sum_{j_c=1}^{m}P_I\_DSM(i_c,j_c),\sum_{j_c=1}^{m}P_J\_DSM(i_c,j_c))}. \tag{2}$$

3.2.3 Project structural similarity

According to Eqs. (1) and (2), the project structural similarity coefficient based on the intra-project knowledge transfer relationship can be measured as follows:

$$S_{kt}(P_I,P_J)=S_{kt_D}(P_I,P_J)+S_{kt_{ID}}(P_I,P_J). \tag{3}$$

3.3 Measuring project attribute similarity based on knowledge increments

During NPD, project knowledge is continuously developed. Knowledge increment refers to the process through which a project gains new knowledge from previous activities, specifically through knowledge learning that results from activity iteration (Ochodek et al., 2019). This study measures knowledge increments based on both the sequential and iterative development of the activities within each project.

3.3.1 Sequential-based knowledge increments

The sequential-based knowledge increment refers to increasing knowledge by receiving upstream activity information. The sequential-based knowledge increment of each activity can be shown in the sub-diagonal DSM of a project. The existing method of measuring knowledge increment is mainly related to the duration devoted to each activity (Carrillo and Franza 2006). Further, Özkan-Seely et al. (2015) proposed an incremental knowledge development function, that is $\gamma(t)[D(t)]^{\rho_1}$, related to the changing rate of knowledge development (i.e., $\gamma(t)$), the team's current knowledge level (i.e., $D(t)$) and the diminishing rate of return (i.e., ρ_1). According to the knowledge development function proposed by Özkan-Seely et al. (2015), we assume that the capability of personnel (e.g., the workload of each person in a day) and immediate work for similar activities in different projects ensure that the activity durations are within a reasonable range. Thus, we propose the sequential-based knowledge increment model related to the duration of the activity, representing the knowledge level of activity m of the project P_I:

$$P_I\_KD(m)=\mu_m\ln(D_m+1), \tag{4}$$

where D_m is the duration of activity m, $\mu_m\in(0,1)$ is the changing rate of knowledge maturity, and its value is affected by factors such as the quality and quantity of personnel allocated to the project activity. In this study, to normalize the value of P_I_KD, we determined the value of μ_m according to the actual situation of the project and set it as a constant.

3.3.2 Iteration-based knowledge increments

Iteration-based knowledge increments refer to an increase in knowledge through the receipt of feedback information. The iteration-based knowledge increments of each activity are shown in the super diagonal DSM of a project. Existing research on activity iteration mainly focuses on rework risk but ignores its influence on knowledge increments. Rework risk can be calculated using the rework probability and impact (Zou et al., 2024). Therefore, we present an iteration-based knowledge increment model related to the probability of iteration (*POI*), iteration impact strength (*IIS*), and the number of iterations (*NOI*). An example of this is shown in Fig. 4.

First, we integrate the iteration probability DSM and iteration impact strength DSM to measure knowledge increments generated by the iteration points corresponding to the ith row and jth column of the super-diagonal of the DSM. As shown in Fig. 4 (b), the super-diagonal elements of probability DSM represent the probability that the information of the downstream activity is iterated to the upstream activity. The sub-diagonal elements of probability DSM represent the probability that after the upstream activity is iterated, its information will be passed to the downstream activity and iterated again. For example, $POI(2,4)=0.6$ indicates the probability that the information of activity A_4 is iterated to activity A_2 is 60%. Figure 4(c) shows that DSM's impact strength refers to the percentage of iteration-based knowledge increments to sequential-based knowledge increments. For example, $IIS(2,4)=0.5$ means that activity A_4 makes activity A_2 generate iterative knowledge, and the iteration-based knowledge increment amount of activity A_2 accounts for 50% of the sequential-based knowledge increment. The iteration-based knowledge increments influenced by the probability and impact strength from activities j to i can be calculated as follows:

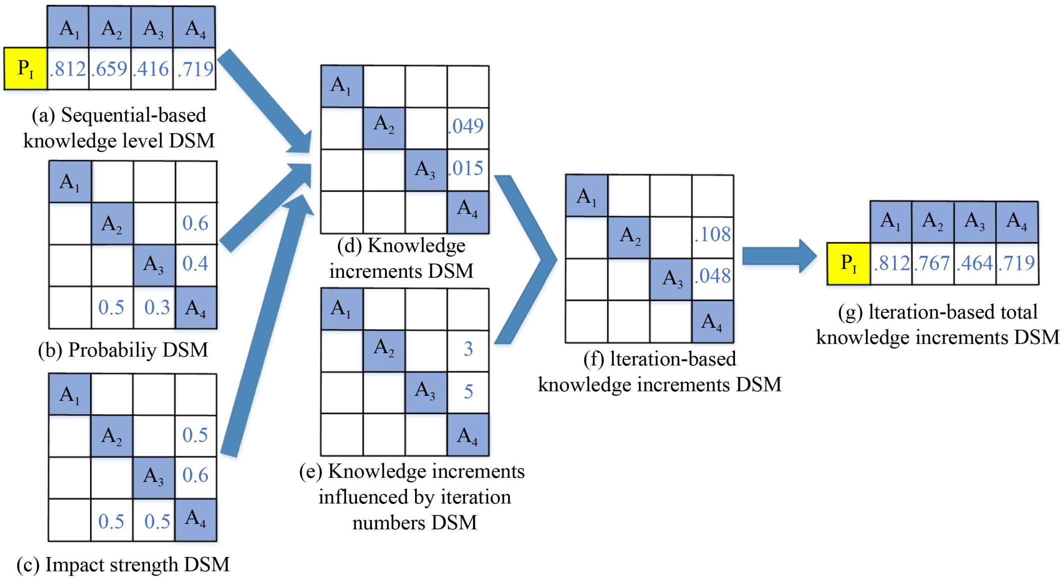


Fig. 4 An example of calculating the knowledge increments-related iteration.

$$KII_P(i,j) = POI(i,j)\times IIS(i,j)\times \sum_{m=i}^{n} P_{I\,KD(m)} \times POI(m,i)\times IIS(m,i), \quad (5)$$

where n is the number of project activities, $POI(i,j)\times IIS(i,j)$ represents the increased value of knowledge caused by direct iteration, and $\sum_{m=i}^{n} P_I\_KI(m)\times POI(m,i)\times IIS(m,i)$ represents the increased value of knowledge generated by indirect iteration.

The NPD project improves gradually through multiple iterations; therefore, the number of iterations should be considered (Wu, 2015). As the number of iterations increase, the rate of iteration-based knowledge increments gradually decreases, and the total amount of iterative knowledge of the activity increases spirally, similar to the trend in knowledge acquisition ability (Wu, 2015). Therefore, the rate of decreased knowledge with iteration number (ROK_N) can be approximated by Eq. (6):

$$ROK_N(i,j) = \alpha_0 \times \exp(\beta_0 \times (1-N)) + \delta, \quad (6)$$

where N=1,2...i represents the number of activity iterations. α_0 and β_0 are the knowledge learning ability parameters; and δ is the correction parameter.

Therefore, we use the KII_N to measure the impact of the number of iterations on iteration-based knowledge increments corresponding to the ith row and jth column of the super-diagonal DSM, as calculated using Eq. (7):

$$KII_N(i,j) = \sum_{N=1}^{n} ROK_N(i,j). \quad (7)$$

Furthermore, since KII_N is directly proportional to KII_P, Eq. (8) reflects the relative value of iteration-based knowledge increments at each iteration point:

$$KII(i,j) = \mu_0 \times KII_P(i,j)\times KII_N(i,j), \quad (8)$$

where μ_0 is a constant greater than 0.

Finally, the total iteration-based knowledge increments generated by the iteration of activity i in the project P_I can be obtained as follows:

$$P_I\_KI(i) = \sum_{j=i+1}^{n} KII(i,j). \quad (9)$$

For example, according to Figs. 4(a)–4(c), we obtain Fig. 4(d). $KII_P(2,4)$ = 0.6 × 0.5 × 0.659 × 0.5 × 0.5 = 0.049, $KII_P(3,4) = 0.4\times0.6\times0.416\times0.3\times0.5 = 0.015$. Furthermore, by combining Fig. 4(e) with Eqs. (6)–(7), we obtain $KII_N(2,4) = 2.199$, $KII_N(3,4) = 3.201$. Then, multiplying the corresponding data in Figs. 4(d) and 4(e), Fig. 4(f) can be obtained. $KII(2,4) = 0.049\times2.199 = 0.108$, $KII(3,4) = 0.015\times3.201 = 0.048$. Finally, by adding the corresponding data in Figs. 4(a) and 4(f), Fig. 4(g) is obtained. $P_I\_KI(2) = 0.659+0.108 = 0.767$, $P_I\_KI(3) = 0.416+0.048 = 0.464$.

3.3.3 Project attribute similarity

According to the above analysis, we present the total knowledge increments for each activity of the project P_I by adding P_I_KD and P_I_KI:

$$P_I\_KA(i) = P_I\_KD(i) + P_I\_KI(i). \quad (10)$$

Then, to achieve the most intuitive effect, we use the Euclidean method, representing the straight-line distance, to calculate the similarity coefficient between each pair of projects. According to the knowledge increments of each project's activities in Fig. 5(a), the similarity distance between each pair of projects can be calculated as follows:

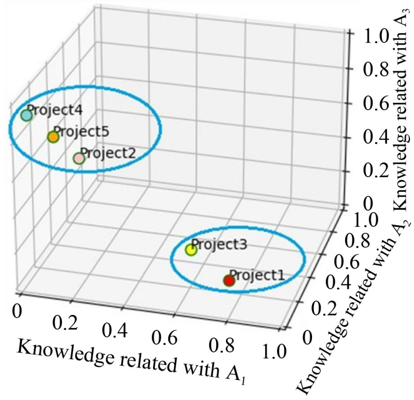
$$D_{ks}(P_I,P_J) = D_{ks}(P_J,P_I) = \sqrt{\sum_{a=1}^{N}(P_I\_KA(a) - P_J\_KA(a))^2}, \quad (11)$$

	A_1	A_2	A_3
P_1	0.8	0	0.2
P_2	0.2	0.1	0.7
P_3	0.6	0.2	0.2
P_4	0	0.1	0.9
P_5	0.1	0.1	0.8

(a) Knowledge increments of each project's activities

	P_1	P_2	P_3	P_4	P_5
P_1	1	0.56	0.78	0.48	0.52
P_2	0.56	1	0.61	0.78	0.88
P_3	0.78	0.61	1	0.52	0.56
P_4	0.48	0.78	0.52	1	0.88
P_5	0.52	0.88	0.56	0.88	1

(b) Project attribute similarity matrix



(c) 3D (three activities) visualization graph of five projects based on knowledge

Fig. 5 A hypothetical example of quantifying project attribute similarity matrix (5 projects and 3 activities).

where a=1,2,3,...,N indicates the activities.

Further, we measured the project attribute similarity of multiple projects' activities as follows:

$$D'_{ks}(P_I,P_J)=D'_{ks}(P_J,P_I)=\frac{1}{(1+D_{ks}(P_I,P_J))}. \qquad (12)$$

In Eq. (12), a big $D'_{ks}(P_I,P_J)$ means a high project attribute similarity between each pair of projects.

Based on the knowledge increments of each project's activities, we can calculate the matrix of the project attribute similarity coefficient, which is a symmetric matrix. In other words, the values of the super-diagonal and sub-diagonal are numerically equal (Fig. 5(b)).

For visualization, we use an example with only three activities in a three-dimensional space (Fig. 5(c)). Projects P_1 and P_3 were close to each other, representing a group of similar projects. Similarly, projects P_2, P_4, and P_5 were close to each other, representing another similar project group. The clustering situation shown in Fig. 5(c) is consistent with the results of the similarity matrix obtained using the Euclidean method (Fig. 5(b)). Therefore, to handle situations with multiple activities, we use the distance similarity method to improve the accuracy of measuring project attribute similarity.

Using the above models, we obtain a project similarity matrix from the perspective of knowledge management.

3.4 Clustering analysis of NPD Projects

3.4.1 Integrated similarity combined structural and attribute similarity

To provide a more comprehensive approach for measuring project similarity, we combine two similarity coefficients: structural similarity based on intra-project knowledge transfer and attribute similarity based on knowledge increments. Therefore, the similarity between projects P_I and P_J can be measured using the following equation:

$$SP(P_I,P_J)=\omega_{kt}\times S_{kt}(P_I,P_J)+\omega_{ks}\times D'_{ks}(P_I,P_J), \qquad (13)$$

where ω_{kt} and ω_{ks} represent the weight of structural similarity and attribute similarity, respectively, determined by the project manager according to the characteristics of multiple projects. $\omega_{kt}+\omega_{ks}=1$.

After obtaining the integrated similarity coefficients, we can cluster projects based on their similarity. This clustering analysis enhances knowledge utilization and resource productivity, while also reducing overall costs and time.

3.4.2 Hierarchical clustering method

Based on the calculated similarity between projects, relevant clustering algorithms can be used to cluster multiple projects. The bottom-up agglomerative method is a commonly used hierarchical clustering method (Daie and Li, 2016). This method starts by treating each element as an individual group, then identifies similar elements based on linkage criteria, continuing until the specified conditions are met and a cohesive cluster is formed. The advantages of this method are as follows: (1) it does not require pre-determining the number of clusters, and (2) it provides effective visualization and is easy to interpret (Patriarca et al., 2023). Therefore, this paper applies the bottom-up agglomerative approach to cluster multiple projects and uses the average linkage clustering algorithm to find the solution.

Additionally, to select reasonable clustering results, this study uses the Cophenetic correlation coefficient (Kumar and Toshniwal, 2016) and the Silhouette index (Gao and Wu, 2019) to evaluate the clustering outcomes.

4 Experiments and analysis of results

This study utilizes an industrial multi-project example to illustrate the proposed concepts and models. We examine a multi-project scenario from a Chinese technology company, which includes 14 projects: $\{P_1, P_2, P_3, P_4, P_5, P_6, P_7, P_8, P_9, P_{10}, P_{11}, P_{12}, P_{13}, P_{14}\}$. The purpose of these projects is to develop various models of mobile phones, each comprising several development activities. The activities have both sequential development relationships and iterative relationships. Additionally, there may also be several common activities shared between the projects. This type of product development project is suitable for validating the proposed model.

Table 1 provides detailed information on the multi-projects, including the activities for each project, the duration of these activities and the number of the iterations between the activities.

Table 1 Duration and number of iterations of projects' activities

Projects	Activities	Durations (days)	Number of Iterations	Projects	Activities	Durations (days)	Number of Iterations
P_1	A_1	14		P_8	A_1	15	
	A_2	10			A_2	10	
	A_6	8			A_5	4	
	A_7	15	$A_7 \to A_6$:5		A_6	10	
					A_7	16	$A_7 \to A_6$:5
P_2	A_1	10		P_9	A_1	16	
	A_2	3			A_2	10	
	A_3	5	$A_3 \to A_2$:3		A_3	3	
	A_4	12			A_5	3	
	A_5	18	$A_5 \to A_4$:3		A_6	9	
	A_6	20			A_7	16	$A_7 \to A_6$:1
P_3	A_1	3		P_{10}	A_1	3	
	A_2	8			A_2	17	
	A_5	5			A_4	5	
	A_6	3	$A_6 \to A_2$:5		A_5	12	
			$A_6 \to A_5$:2		A_6	3	$A_6 \to A_2$:3
					A_7	5	
P_4	A_1	3		P_{11}	A_1	10	
	A_2	21			A_4	15	
	A_3	5			A_5	18	
	A_5	7			A_6	19	$A_6 \to A_5$:3
	A_6	3	$A_6 \to A_5$:2	P_{12}	A_1	16	
	A_7	6			A_2	10	
P_5	A_1	10			A_3	3	
	A_2	3			A_4	3	
	A_4	16			A_5	3	
	A_5	18	$A_5 \to A_4$:5		A_6	8	
	A_6	21	$A_6 \to A_5$:4		A_7	14	$A_7 \to A_6$:1
P_6	A_1	3		P_{13}	A_1	16	
	A_2	21			A_2	10	
	A_5	12			A_5	3	
	A_7	5			A_6	11	
					A_7	16	$A_7 \to A_6$:4
P_7	A_1	16		P_{14}	A_1	9	
	A_2	10			A_3	6	
	A_4	3			A_4	15	
	A_6	11	$A_6 \to A_2$:5		A_5	20	
	A_7	14	$A_7 \to A_6$:5		A_6	20	$A_6 \to A_5$:3

The activities are named as follows: Activity A_1 (customer requirements analysis), Activity A_2 (chip design), Activity A_3 (memory design), Activity A_4 (screen design), Activity A_5 (battery design), Activity A_6 (product verification), and Activity A_7 (product trial operation). In this study, key factors influencing project similarity include the sequential development relationships, iterative relationships between activities, and the number of shared activities.

We interviewed the program manager, project manager, and other core project members of the firm. The following key questions were investigated during the interviews: (1) How can knowledge be defined and measured in practical projects? (2) To what extent does the enterprise use knowledge to manage multiple projects? How can enterprises realize knowledge transfer? (3) How can we understand common activities and identify relationships among projects in practice? How do iterations affect inter-project knowledge transfer? And so on. According to the responses provided by the interviews, we found that the enterprise realized the importance of knowledge transfer among multiple projects. Therefore, it is necessary to study project clustering by identifying the similarity of knowledge among projects through common activities. Additionally, projects with similar knowledge transfer relationships should be grouped to enhance communication, coordination, and knowledge transfer.

We present the knowledge transfer relationship graphs of these multi-projects, as shown in Fig. 6, an extended application of DSM in the multi-project domain. It indicates knowledge transfer relationships from two levels: one is the intra-project knowledge transfer relationships based on the actual condition of the 14 projects

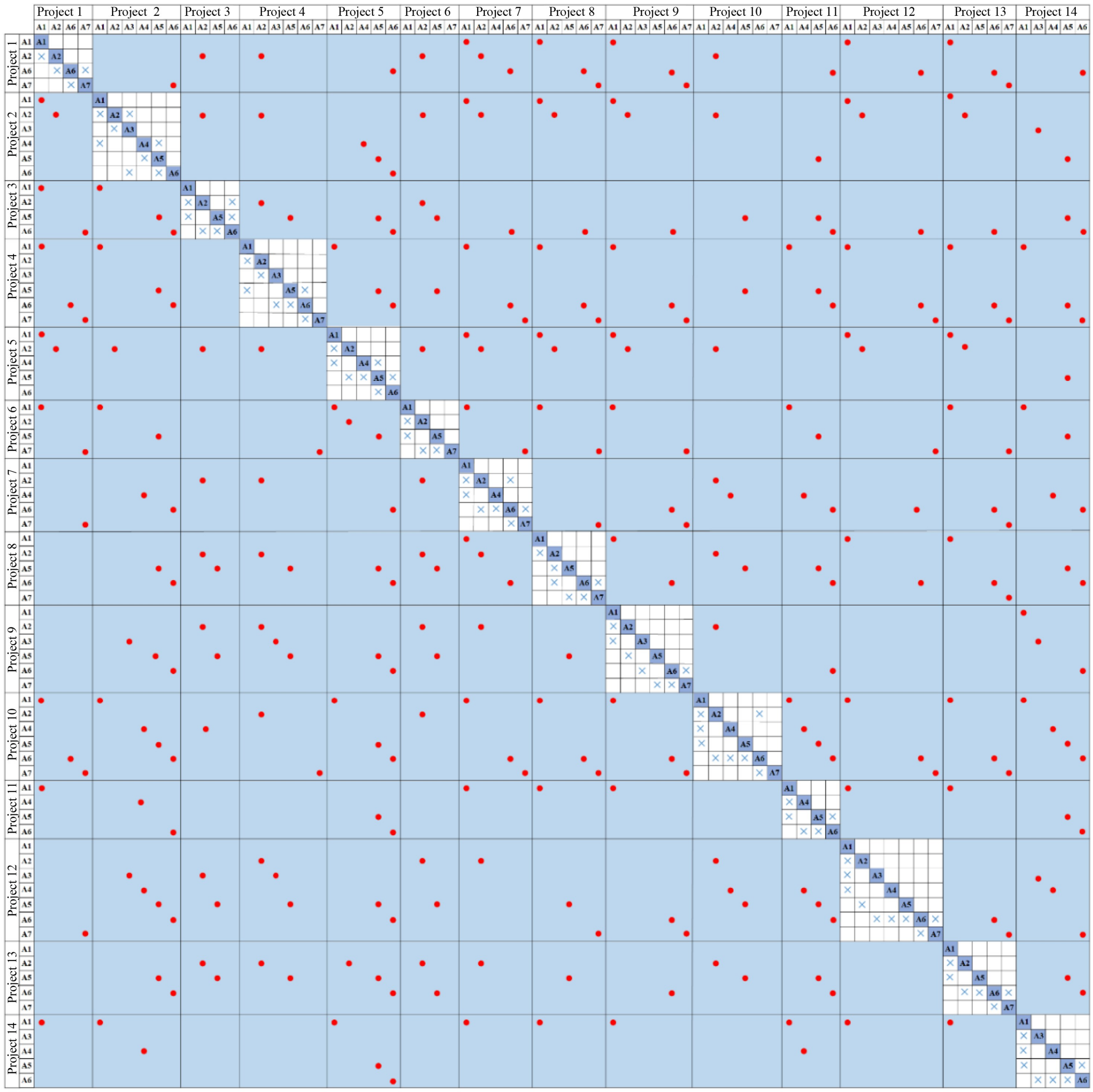


Fig. 6 An example of representing the knowledge transfer among projects.

(diagonal matrices). The other is the knowledge transfer between each pair of projects by comparing the amount of each project's knowledge increment based on sequence and iteration (off-diagonal matrices). As shown in Fig. 6, each diagonal matrix shows the information of intra-project knowledge transfer relationships. The sub-diagonal marks denote the sequential-based knowledge transfer relationships, and the super-diagonal marks indicate the iteration-based knowledge transfer relationships. Then, using Eqs. (4)–(10), we measured the total knowledge increment of each activity. In Eqs. (4)–(8), $\mu_m = 0.3$, α_0=0.75, $\beta_0 = 0.18$, $\delta = 0.1$, $\mu_0 = 1$, $\omega_{kt} = 0.3$, and $\omega_{ks} = 0.7$ are evaluated through simulation experiments and project managers' experience. The dots in the off-diagonal matrices represent the knowledge transfer among projects (i.e., the red dots in the upper-right part of the DSM represent the knowledge transfer from the column to the row project. The red dots in the lower-left part of the DSM represent the knowledge transfer from the row project to the column project). Therefore, activities iteration and knowledge transfer among multiple projects in one DSM can be expressed, enhancing visibility in multi-project management.

Next, based on the intra-project knowledge transfer relationships (indicated by the marks in the diagonal matrix in Fig. 6) and Eqs. (1)–(3), we calculated the project structural similarity using Matlab software (Table 2(a)). Furthermore, using the data in Table 1 and

Table 2 The calculation results of the project similarity matrix

	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}	P_{11}	P_{12}	P_{13}	P_{14}
P_1	1	0.37	0.44	0.50	0.43	0.43	0.67	0.80	0.60	0.46	0.00	0.58	0.80	0.00
P_2	0.37	1	0.46	0.56	0.62	0.37	0.46	0.23	0.36	0.53	0.54	0.50	0.50	0.53
P_3	0.44	0.46	1	0.75	0.64	0.50	0.60	0.38	0.25	0.71	0.67	0.43	0.61	0.54
P_4	0.50	0.56	0.78	1	0.61	0.63	0.42	0.40	0.56	0.60	0.67	0.59	0.64	0.69
P_5	0.43	0.62	0.64	0.61	1	0.44	0.46	0.38	0.33	0.53	0.69	0.50	0.54	0.60
P_6	0.43	0.37	0.50	0.63	0.44	1	0.37	0.70	0.67	0.60	0.40	0.47	0.67	0.33
P_7	0.67	0.46	0.60	0.42	0.46	0.37	1	0.89	0.46	0.80	0.44	0.56	0.50	0.40
P_8	0.80	0.23	0.38	0.40	0.38	0.70	0.89	1	0.77	0.41	0.12	0.62	0.71	0.10
P_9	0.60	0.36	0.25	0.56	0.33	0.67	0.46	0.77	1	0.37	0.12	0.76	0.61	0.50
P_{10}	0.46	0.53	0.71	0.60	0.53	0.60	0.80	0.41	0.37	1	0.61	0.67	0.67	0.67
P_{11}	0.00	0.54	0.67	0.67	0.69	0.40	0.44	0.12	0.12	0.61	1	0.54	0.44	0.83
P_{12}	0.58	0.50	0.43	0.59	0.50	0.47	0.56	0.62	0.76	0.67	0.54	1	0.71	0.73
P_{13}	0.80	0.50	0.61	0.64	0.54	0.67	0.50	0.71	0.61	0.67	0.44	0.71	1	0.40
P_{14}	0.00	0.53	0.54	0.67	0.60	0.33	0.40	0.10	0.50	0.67	0.83	0.73	0.40	1

(a) project structural similarity matrix

	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}	P_{11}	P_{12}	P_{13}	P_{14}
P_1	1	0.38	0.45	0.48	0.38	0.46	0.70	0.67	0.63	0.47	0.37	0.58	0.70	0.36
P_2	0.38	1	0.45	0.44	0.64	0.39	0.41	0.41	0.43	0.47	0.58	0.48	0.41	0.67
P_3	0.45	0.45	1	0.56	0.44	0.59	0.43	0.49	0.46	0.57	0.42	0.45	0.48	0.40
P_4	0.48	0.44	0.56	1	0.40	0.59	0.45	0.53	0.57	0.57	0.38	0.54	0.52	0.40
P_5	0.38	0.64	0.44	0.40	1	0.39	0.41	0.41	0.40	0.49	0.70	0.44	0.41	0.58
P_6	0.46	0.39	0.59	0.59	0.39	1	0.44	0.50	0.47	0.59	0.37	0.46	0.49	0.36
P_7	0.70	0.41	0.43	0.45	0.41	0.44	1	0.61	0.58	0.49	0.40	0.63	0.63	0.38
P_8	0.67	0.41	0.49	0.53	0.41	0.50	0.61	1	0.70	0.53	0.40	0.63	0.93	0.38
P_9	0.63	0.43	0.46	0.57	0.40	0.47	0.58	0.70	1	0.49	0.39	0.70	0.71	0.40
P_{10}	0.47	0.47	0.57	0.57	0.49	0.59	0.49	0.53	0.49	1	0.44	0.53	0.51	0.42
P_{11}	0.37	0.58	0.42	0.38	0.70	0.37	0.40	0.40	0.39	0.44	1	0.42	0.40	0.63
P_{12}	0.58	0.48	0.45	0.54	0.44	0.46	0.63	0.63	0.70	0.53	0.42	1	0.63	0.43
P_{13}	0.70	0.41	0.48	0.52	0.41	0.49	0.63	0.93	0.71	0.51	0.40	0.63	1	0.38
P_{14}	0.36	0.67	0.40	0.40	0.58	0.36	0.38	0.38	0.40	0.42	0.63	0.43	0.38	1

(b) project attribute similarity matrix

(*Continued*)

	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}	P_{11}	P_{12}	P_{13}	P_{14}
P_1	1	0.38	0.45	0.48	0.39	0.45	0.69	0.71	0.62	0.47	0.26	0.58	0.73	0.25
P_2	0.38	1	0.45	0.48	0.64	0.38	0.43	0.36	0.41	0.49	0.57	0.48	0.44	0.62
P_3	0.45	0.45	1	0.61	0.50	0.57	0.48	0.46	0.40	0.61	0.49	0.44	0.52	0.44
P_4	0.48	0.48	0.61	1	0.47	0.61	0.44	0.50	0.57	0.58	0.47	0.55	0.56	0.49
P_5	0.39	0.64	0.50	0.47	1	0.41	0.43	0.40	0.38	0.50	0.69	0.46	0.45	0.58
P_6	0.45	0.38	0.57	0.61	0.41	1	0.42	0.56	0.53	0.60	0.38	0.46	0.54	0.35
P_7	0.69	0.43	0.48	0.44	0.43	0.42	1	0.69	0.54	0.69	0.41	0.61	0.59	0.38
P_8	0.71	0.36	0.46	0.50	0.40	0.56	0.69	1	0.72	0.49	0.32	0.63	0.87	0.30
P_9	0.62	0.41	0.40	0.57	0.38	0.53	0.54	0.72	1	0.46	0.31	0.72	0.68	0.43
P_{10}	0.47	0.49	0.61	0.58	0.50	0.6	0.69	0.49	0.46	1	0.49	0.57	0.56	0.49
P_{11}	0.26	0.57	0.49	0.47	0.69	0.38	0.41	0.32	0.31	0.49	1	0.46	0.41	0.69
P_{12}	0.58	0.48	0.44	0.55	0.46	0.46	0.61	0.63	0.72	0.57	0.46	1	0.65	0.52
P_{13}	0.73	0.44	0.52	0.56	0.45	0.54	0.59	0.87	0.68	0.56	0.41	0.65	1	0.39
P_{14}	0.25	0.62	0.44	0.49	0.58	0.35	0.38	0.30	0.43	0.49	0.69	0.52	0.39	1

(c) project similarity matrix

Eqs. (4)–(12), we calculated the project attribute similarity (Table 2(b)). Subsequently, based on Tables 2(a) and 2(b) and Eq. (13), we obtained the project similarity matrix (Table 2(c)). Finally, we applied the hierarchical clustering algorithm using Python 3.0 to obtain the clustering results.

4.1 Clustering analysis

4.1.1 Clustering results

Figure 7(a) shows a solid line across the cluster tree to distinguish between three programs: Program 1 [P_2, P_5, P_{11} and P_{14}], Program 2 [P_1, P_7, P_8, P_9, P_{12} and P_{13}] and Program 3 [P_3, P_4, P_6 and P_{10}]. The hierarchical clustering heat map is shown in Fig. 7(b).

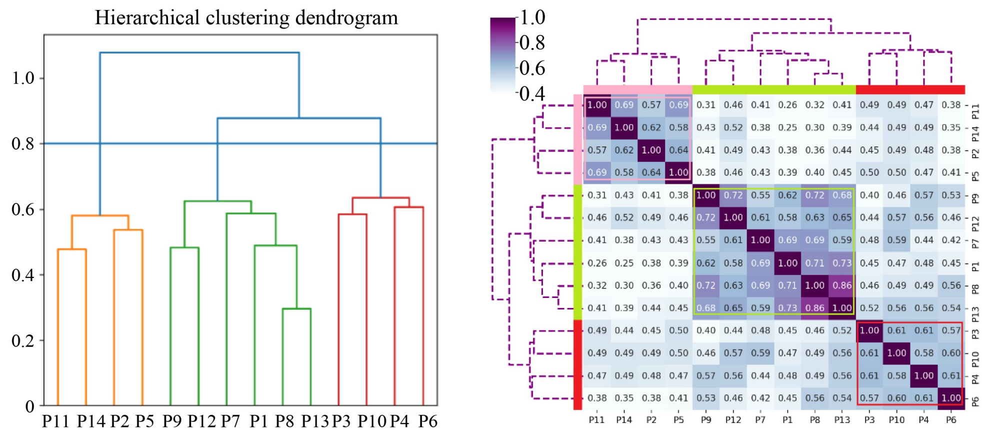
Table 3 shows that agglomerative hierarchical clustering using the average linkage method generates the highest Cophenetic correlation coefficient compared to the other six distance metric approaches.

The clustering results can be evaluated using the Silhouette index (Table 4). A high Silhouette index indicates good clustering performance. Based on this, we determine the optimal number of clusters for our proposed method to be three, dividing the 14 projects into three categories.

4.1.2 The influence of attribute similarity on clustering results

Project attribute similarity has a significant influence on the clustering results. Typically, projects with high attribute similarity facilitate knowledge transfer. Thus, their structural similarity is high. However, in some cases, the project attribute similarity based on total knowledge increment and the project structural similarity based on the knowledge transfer relationships may be inconsistent. Figure 8 shows the clustering results when $\omega_{kt} = 0.75$ and $\omega_{ks} = 0.25$, indicating that the importance of the structural dimension is higher than that of the attribute dimension. The clustering result is as follows: Program G1 [$P_2, P_3, P_4, P_5, P_{10}, P_{11}$, and P_{14}], Program G2 [P_1, P_7, and P_8], and Program G3 [P_6, P_9, P_{12}, and P_{13}]. For example, in Program G1 [$P_2, P_3, P_4, P_5, P_{10}, P_{11}$, and P_{14}], even though project P_{14} has low attribute similarity with projects P_3, P_4, and P_{10}, they are still grouped in the same program due to strong knowledge transfer relationships. When we increase the weight of attribute similarity (i.e., from $\omega_{ks} = 0.25$ to $\omega_{ks} = 0.7$), Program G1 no longer contains projects P_3, P_4, and P_{10}. Program G2 and part of Program G3 form Program 2.

From the above results, we conclude that the implications of program clustering are based on project similarity across several dimensions. Grouping projects with a high degree of similarity into programs facilitates knowledge transfer, thus reducing the costs associated with knowledge transfer between different projects. The proposed clustering approach helps project managers categorize multiple projects into distinct groups (or clusters). Within each group, the experience and knowledge gained from one project can be transferred to the next, enabling the organization to avoid redundant efforts and enhance knowledge transfer efficiency. Additionally, clustering analysis can reveal the specific resource or skill requirements for certain projects, which can facilitate more effective resource allocation.



(a) Cluster tree (b) Hierarchical clustening heat map

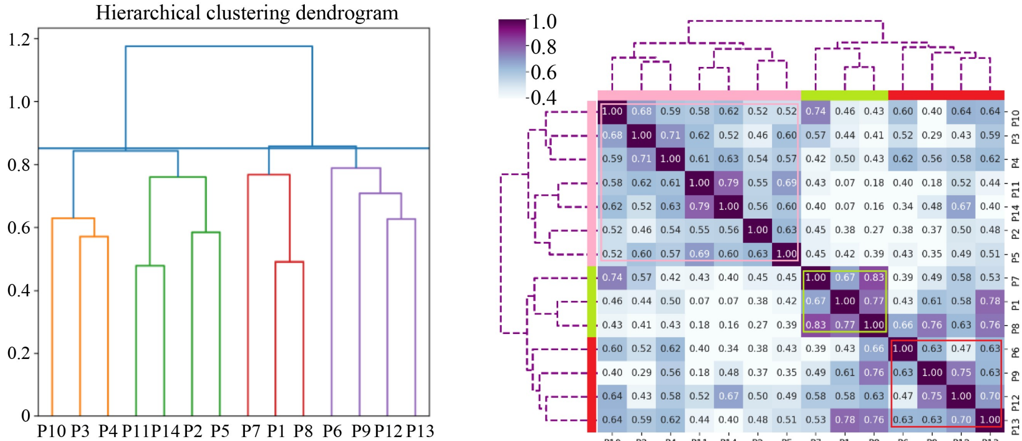
Fig. 7 Visualized results using hierarchical clustering.

Table 3 Comparison of Cophenetic correlation coefficient with different distance metrics

	Ward	Single	Average	Complete	Median	Centroid	Weighted
Cophenetic correlation coefficient	0.867	0.860	0.873	0.869	0.866	0.864	0.871

Table 4 Comparison of Silhouette index with different clustering results

	Results of our proposed method	Alternative 1	Alternative 2	Alternative 3
# Clusters	3	2	4	5
Silhouette index	0.347	0.342	0.258	0.207



(a) Cluster tree (b) Hierarchical clustening heat map

Fig. 8 Results when the structural dimension is higher than the attribute dimension.

4.2 Comparison tests

We used Project 4 (P_4) as an example to assess the influence of indirect connections between common activities and iterative relationships on the project similarity coefficient. By combining the intra-project knowledge transfer relationships (the marks in the diagonal matrix in Fig. 6) with different equations, we obtained the comparison results of the similarity coefficient between P_4 and the other projects (Table 5). In Table 5, we use the abbreviations B, S1, S2, and S3 to represent the baseline values, Situation1, Situation2 and Situation3, respectively. The different values of Baseline, Situation1, Situation2 and Situation3 represent direct knowledge transfer (i.e., traditional equation without iteration), iteration-based direct knowledge transfer (i.e., Eq. (1)), direct and indirect knowledge transfer (i.e., Eq. (3) without iteration), and iteration-based and direct and indirect knowledge transfer (i.e., Eq. (3)), respectively.

We plotted a line chart based on Table 5 (Fig. 9) to

Table 5 Comparison results of similarity coefficient between P_4 and other projects

	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}	P_{11}	P_{12}	P_{13}	P_{14}
B	0.50	0.73	0.67	1	0.40	0.50	0.50	0.33	0.46	0.67	0.57	0.62	0.80	0.55
S1	0.40	0.50	0.67	1	0.46	0.44	0.33	0.27	0.38	0.53	0.68	0.47	0.57	0.62
S2	0.63	0.82	0.78	1	0.60	0.71	0.63	0.58	0.69	0.75	0.57	0.77	0.90	0.64
S3	0.50	0.56	0.78	1	0.61	0.63	0.42	0.40	0.56	0.60	0.67	0.59	0.64	0.69

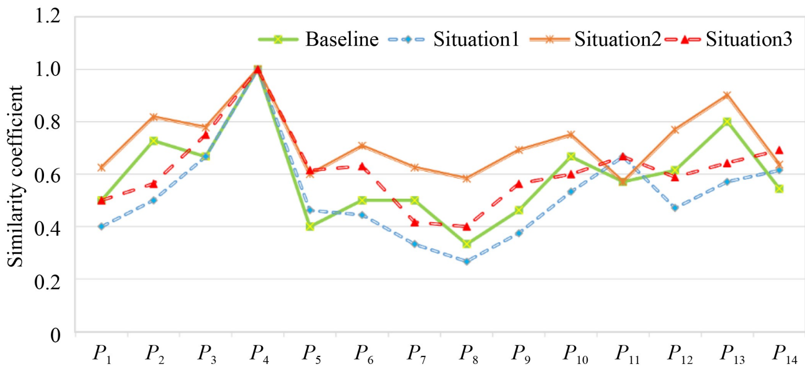


Fig. 9 Similarity coefficient between P_4 and other projects.

facilitate comparative analysis. There are several important considerations in these results shown in Fig. 9.

(1) Iteration-based direct knowledge transfer (Situation 1) has a significant impact on project similarity. For example, the similarity coefficient of the sequential-based knowledge transfer relationship between projects P_4 and P_{13}, based on their sequential knowledge transfer relationship, is most influenced by iteration-based knowledge transfer (i.e., the iterative knowledge transfer between P_4 and P_{13} occurs from activities A_6 to A_5 and A_7 to A_6, respectively). Therefore, the project similarity influenced by the iteration-based direct knowledge transfer relationship is significantly reduced (Baseline and Situations 1, 2, and 3 in Fig. 9). Consequently, the knowledge transfer relationship between projects P_4 and P_{13} is weak, so they are managed in different programs. On the other hand, strong knowledge transfer relationships within a program can enhance knowledge sharing and minimize unnecessary investments. Therefore, iteration-based direct knowledge transfer relationships (i.e., Eq. (1)) should not be overlooked when assessing project similarity.

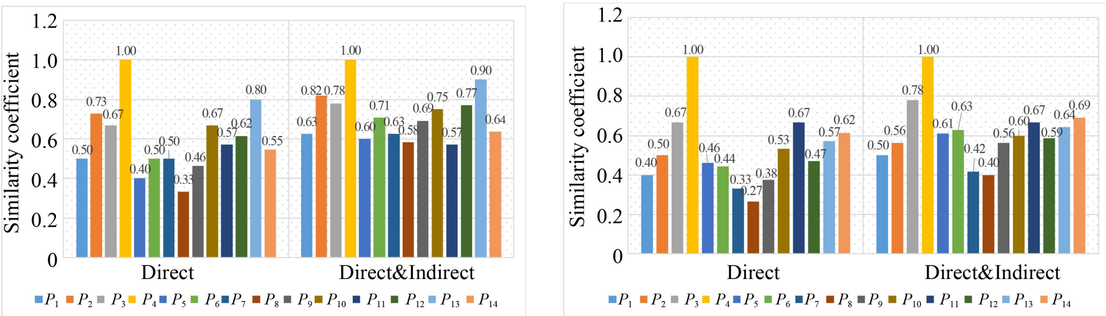
(2) Indirect knowledge transfer relationships within an intra-project (Situation 2 and 3) can reinforce the similarity of interdependent projects. Therefore, projects with strong knowledge transfer relationships should be grouped into the same program. For example, Fig. 10 shows that the similarity coefficient between P_6 and P_4 is equal to P_7 and P_4. At the same time, P_6 is more similar to P_4 than P_7 when considering the intra-project indirect knowledge transfer relationship of common activities. Therefore, there are two indirect knowledge transfer relationships between projects P_4 and P_6 (i.e., $A_2 \rightarrow A_3 \rightarrow A_6 \rightarrow A_7$ and $A_5 \rightarrow A_6 \rightarrow A_7$) and only one between projects P_4 and P_7 (i.e., $A_2 \rightarrow A_3 \rightarrow A_6$). The results also show that the intra-project indirect knowledge transfer relationship between each pair of common activities can affect the degree of project structural similarity, as $S_{as\_D}(P_3, P_4) = S_{as\_D}(P_{11}, P_4)$, while $S_{as}(P_3, P_4) > S_{as}(P_{11}, P_4)$. In other words, when there are many indirect knowledge transfer relationships among projects, the interaction relationship is strengthened, thereby increasing the project similarity coefficient.

4.3 Discussion

Our research contributes to the literature on NPD multi-project management. First, although existing literature suggests that knowledge transfer between NPD projects can enhance organizational creativity (Korhonen et al., 2016), they ignore proposing a method to facilitate this transfer. This paper addresses this gap by developing a similarity-based clustering approach for NPD multi-projects, which models both knowledge transfer relationships and knowledge increments across projects.

Secondly, existing studies assessed the similarity among projects based on the number of common activities (Xu et al., 2022). This study presents a more comprehensive approach based on a network perspective than previous literature, which encompass a fusion of structural and attribute similarities of the project network. It also takes into account the influence of project network features on determining project similarity.

Thirdly, we propose a method for calculating structural similarity among projects that accounts for both direct and indirect knowledge transfer within projects. This approach explains direct and indirect knowledge transfer



(a) With no iterative relationships

(b) With iterative relationships

Fig. 10 Knowledge transfer relationships comparison (between P_4 and other projects).

relationships by analyzing sequence and iteration between activities. This inclusion reflects their substantial impact on project structural similarity. Additionally, we propose a method for measuring attribute similarity among projects, which integrates both sequential-based and iteration-based knowledge increments.

Moreover, this paper also contributes to the practical management of NPD multi-projects. First, the proposed similarity-based clustering method helps project managers organize multiple projects into distinct programs, thereby fostering knowledge sharing across projects. By enhancing the efficiency of knowledge transfer, project teams can more quickly acquire the necessary knowledge and experience, ultimately shortening development cycles and improving both the innovation and quality of new products.

Secondly, we provide an evaluation method that integrates structural and attribute similarities, enabling project managers to more accurately assess the similarity between projects. This approach enables project managers to better identify and manage similarities, optimizing resource allocation and project portfolio management, and ultimately improving project success rates.

5 Conclusions

This study proposes a similarity-based clustering method for managing complex NPD multi-projects, measuring both structural and attribute similarities among projects from the perspective of knowledge management. First, we extend existing research on similarity coefficients based on activity sequences by incorporating the influence of intra-project indirect and iterative processes for knowledge transfer, thus enabling a more comprehensive measurement of project structural similarity. Second, we propose a more sophisticated view than prior approaches to quantifying project attribute similarity by measuring knowledge increments from both sequential and iterative activity interactions. Third, we use the hierarchical clustering method to group high-similarity projects and form different programs to promote knowledge transfer by synthesizing both the structural and attribute similarity coefficients.

We also propose several potential research opportunities for the future. First, although our method is primarily designed for physical product development, its underlying principles and framework can be extended to a wider range of projects. By adjusting the project similarity calculation methods and clustering criteria, it could be applied to software system development and pharmaceutical development. Second, our method is focused on clustering projects within a single enterprise. A key avenue for future research is to enhance this method, making it more universally applicable across different organizations or industries. Finally, beyond knowledge increments from activities, other project characteristics should be explored to refine the measurement of project attribute similarity coefficients.

Competing Interests The authors declare that they have no competing interests.

Appendix A

Acronyms table of each equation.

Acronyms	Definitions
S_{kt_D}	The project structural similarity coefficient based on the intra-project direct knowledge transfer relationship.
S_{kt_ID}	The project structural similarity coefficient influenced by the intra-project indirect knowledge transfer relationship.
S_{kt}	The project structural similarity coefficient based on the intra-project knowledge transfer relationship.
P_I_KD	The sequential-based knowledge increment model.
POI	Probability of iteration.
IIS	Iteration impact strength.
NOI	Number of iterations.
KII_P	The iteration-based knowledge increments influenced by POI and IIS.
ROK_N	The rate of decreased knowledge with iteration number.
KII_N	The iteration-based knowledge increments influenced by NOI.
KII	The iteration-based knowledge increments at each iteration point.

(*Continued*)

Acronyms	Definitions
P_I_KI	The iteration-based knowledge increments model.
P_I_KA	The knowledge increment model.
D_{ks}	The similarity distance model.
D'_{ks}	The project attribute similarity coefficient based on knowledge increments.
SP	The project similarity model (combined structural and attribute similarity).

References

Abbasi D, Ashrafi M, Ghodsypour S (2020). A multi objective-BSC model for new product development project portfolio selection. Expert Systems with Applications, 162: 113757

Abdolvand N, Albadvi A, Aghdasi M (2015). Performance management using a value-based customer-centered model. International Journal of Production Research, 53(18): 5472–5483

Argote L, Fahrenkopf E (2016). Knowledge transfer in organizations: The roles of members, tasks, tools, and networks. Organizational Behavior and Human Decision Processes, 136: 146–159

Bai L, Zheng K, Wang Z, Liu J (2022). Service provider portfolio selection for project management using a BP neural network. Annals of Operations Research, 308(1-2): 41–62

Bashir H, Ojiako U, Marshall A, Chipulu M, Yousif A (2022). The analysis of information flow interdependencies within projects. Production Planning and Control, 33(1): 20–36

Bi Y, Yang Q, Chang M, Yao T (2020). DSM-based knowledge transfer modeling between projects for multi-project clustering analysis. Proceedings of the 22nd International DSM Conference (DSM 2020), MIT, Cambridge, Massachusetts, USA. 105–113

Browning T R (2016). Design structure matrix extensions and innovations: A survey and new opportunities. IEEE Transactions on Engineering Management, 63(1): 27–52

Cao H, Hall N, Wan G, Zhao W (2024). Optimal intraproject learning. Manufacturing & Service Operations Management, 26(2): 681–700

Carrillo J E, Franza R M (2006). Investing in product development and production capabilities: The crucial linkage between time-to-market and ramp-up time. European Journal of Operational Research, 171(2): 536–556

Chen H, Ding G, Zhang J, Li R, Jiang L, Qin S (2022). A filtering genetic programming framework for stochastic resource constrained multi-project scheduling problem under new project insertions. Expert Systems with Applications, 198: 116911

Daie P, Li S (2016). Managing product variety through configuration of pre-assembled vanilla boxes using hierarchical clustering. International Journal of Production Research, 54(18): 5468–5479

Frank A G, Ribeiro J L D (2014). An integrative model for knowledge transfer between new product development project teams. Knowledge Management Research and Practice, 12(2): 215–225

Gao X, Wu S (2019). CUBOS: An internal cluster validity index for categorical data. Tehnicki Vjesnik, 26(2): 486–494

Haug A (2023). Factors influencing knowledge sharing in new product development in high-tech manufacturing firms. International Journal of Production Research, 61(19): 6418–6433

Klessova S, Thomas C, Engell S (2020). Structuring inter-organizational R&D projects: Towards a better understanding of the project architecture as an interplay between activity coordination and knowledge integration. International Journal of Project Management, 38(5): 291–306

Korhonen T, Laine T, Lyly-Yrjanainen J, Suomala P (2016). Innovation for multiproject management: The case of component commonality. Project Management Journal, 47(2): 130–143

Kumar S, Toshniwal D (2016). Analysis of hourly road accident counts using hierarchical clustering and cophenetic correlation coefficient (CPCC). Journal of Big Data, 3(1): 13–24

Lin L, Wang H (2019). Dynamic incentive model of knowledge sharing in construction project team based on differential game. Journal of the Operational Research Society, 70(12): 2084–2096

Liu T, Shi Z, Dong H, Bai J, Yan Y (2024). Risk evaluation for the task transfer of an aircraft maintenance program based on a multi-element connection number. Frontiers of Engineering Management, 11(1): 16–31

Navaei J, Elmaraghy H (2016). Grouping part/product variants based on networked operations sequence. Journal of Manufacturing Systems, 38: 63–76

Navaei J, Elmaraghy H (2018). Optimal operations sequence retrieval from master operations sequence for part/product families. International Journal of Production Research, 56(1–2): 140–163

Ochodek M, Staron M, Meding W (2019). SimSAX: A measure of project similarity based on symbolic approximation method and software defect inflow. Information and Software Technology, 115: 131–147

Ou Y, Guo Q, Liu J (2022). Identifying spreading influence nodes for social networks. Frontiers of Engineering Management, 9(4): 520–549

Özkan-Seely F, Gaimon C, Kavadias S (2015). Dynamic knowledge transfer and knowledge development for product and process design teams. Manufacturing & Service Operations Management, 17(2): 177–190

Pan M, Chandrasekaran A, Hill J, Rungtusanatham M (2022). Multidisciplinary R&D project success in small firms: The role of multiproject status and project management experience. Production and Operations Management, 31(7): 2806–2821

Patriarca R, Simone F, Di Gravio G (2023). Supporting weather forecasting performance management at aerodromes through anomaly detection and hierarchical clustering. Expert Systems with Applications, 213: 119210

Qiao Y, Fricker J D, Labi S (2019). Quantifying the similarity between different project types based on their pay item compositions: application to bundling. Journal of Construction Engineering and Management, 145(9): 04019053

Santoro F M, Borges M R, Rezende E A (2006). Collaboration and knowledge sharing in network organizations. Expert Systems with Applications, 31(4): 715–727

Stock G N, Tsai J C, Jiang J, Klein G (2021). Coping with uncertainty: Knowledge sharing in new product development projects. International Journal of Project Management, 39(1): 59–70

Vali M, Salimifard K, Gandomi A H, Chaussalet T J (2022). Care process optimization in a cardiovascular hospital: An integration of simulation-optimization and data mining. Annals of Operations Research, 318(1): 685–712

Wu Y (2015). Organizational structure and product choice in knowledge-intensive firms. Management Science, 61(8): 1830–1848

Xu J, He M, Jiang Y (2022). A novel framework of knowledge transfer system for construction projects based on knowledge graph and transfer learning. Expert Systems with Applications, 199: 116964

Yang Q, Yang N, Browning T R, Jiang B, Yao T (2022). Clustering product development project organization from the perspective of social network analysis. IEEE Transactions on Engineering Management, 69(6): 2482–2496

Yang Q, Yao T, Lu T, Zhang B (2014). An overlapping-based design structure matrix for measuring interaction strength and clustering analysis in product development project. IEEE Transactions on Engineering Management, 61(1): 159–170

Zhang L, Zhao M, Wang Q (2016). Research on knowledge sharing and transfer in remanufacturing engineering management based on SECI model. Frontiers of Engineering Management, 3(2): 136–143

Zhang Z, Min M (2022). Research on the NPD coordination, knowledge transfer process and innovation performance of interfirm projects in China. Asia Pacific Journal of Management, 39(4): 1161–1186

Zhou Q, Deng X, Wang G, Mahmoudi A (2022). Linking elements to outcomes of knowledge transfer in the project environment: Current review and future direction. Frontiers of Engineering Management, 9(2): 221–238

Zou X, Yang Q, Wang Q, Jiang B (2024). Measuring the system resilience of project portfolio network considering risk propagation. Annals of Operations Research, 340(1): 693–721